

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-134546

(43)Date of publication of application : 18.05.2001

(51)Int.Cl.

G06F 15/177

G06F 11/20

G06F 12/16

G06F 15/16

(21)Application number : 11-314813

(71)Applicant : NEC CORP

(22)Date of filing : 05.11.1999

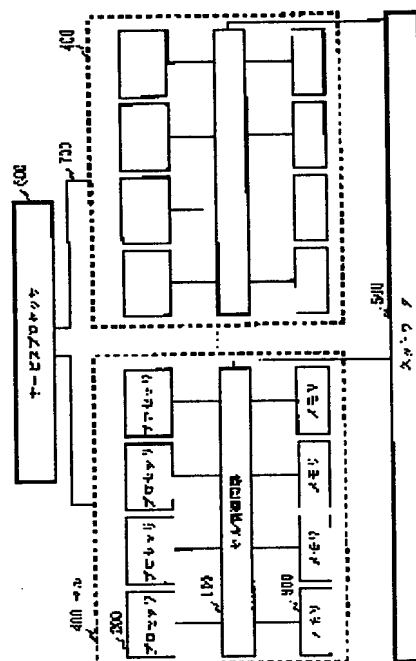
(72)Inventor : KISHIDA YUICHI

(54) FAULT PROCESSING METHOD OF MULTIPROCESSOR SYSTEM

(57)Abstract:

PROBLEM TO BE SOLVED: To improve the reliability and availability of a multiprocessor system, wherein cells including memories and more than one processor are mutually connected by an interconnecting network by preventing a fault from being propagated if the fault occurs to one cell in the operation of the multiprocessor system.

SOLUTION: A cell 400 which detects a fault when accessing a memory 300 sends a fault report to a service processor 600. The service processor 600 broadcasts the received fault report to all cells 40 by returning it as a command temporarily stopping cell operation through hardware to instantaneously stop all the cells 400 from operating. Then the service processor 600 gathers information needed for fault analysis from the respective cells 400 to analyze the fault, disconnects a suspicious cell logically from the system and reconstitutes the system, and resets the temporary stop state of the respective cells to carry on the operation of the system.



LEGAL STATUS

[Date of request for examination] 10.10.2000

[Date of sending the examiner's decision of rejection] 03.09.2002

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2001-134546

(P2001-134546A)

(43)公開日 平成13年5月18日(2001.5.18)

(51)Int.Cl. ⁷	識別記号	F I	テマコード*(参考)
G 0 6 F 15/177	6 7 8	G 0 6 F 15/177	6 7 8 A 5 B 0 1 8
	6 7 2		6 7 2 H 5 B 0 3 4
11/20	3 1 0	11/20	3 1 0 K 5 B 0 4 5
12/16	3 1 0	12/16	3 1 0 C
15/16	6 4 0	15/16	6 4 0 A
審査請求 有 請求項の数4 O L (全 11 頁)			

(21)出願番号 特願平11-314813

(22)出願日 平成11年11月5日(1999.11.5)

(71)出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72)発明者 岸田 裕一

東京都港区芝五丁目7番1号 日本電気株式会社内

(74)代理人 100088959

弁理士 境 廣巳

Fターム(参考) 5B018 GA06 KA02

5B034 BB11 BB15 CC01 DD04

5B045 BB04 HH01 JJ09 JJ37 JJ38

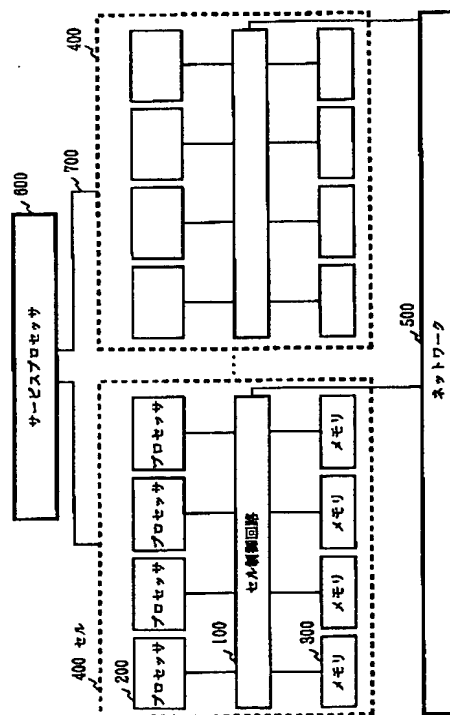
JJ45 JJ46 JJ48

(54)【発明の名称】 マルチプロセッサシステムの障害処理方法

(57)【要約】

【課題】 各々メモリと1つ以上のプロセッサを含む複数のセルが相互結合網によって相互に接続されたマルチプロセッサシステムの運用中に或るセルで障害が発生した時、その障害の伝搬を未然に防止し、システムの信頼性、可用性を向上させる。

【解決手段】 メモリ300 のアクセス時に障害を検出したセル400 は、サービスプロセッサ600 へ障害報告を送出する。サービスプロセッサ600 は、受信した障害報告を、セル動作を一時停止する指令としてハードウェア的に折り返して全セル400 にブロードキャストすることで、全セル400 の動作を瞬時に停止させる。その後サービスプロセッサ600 は、各セル400 から障害解析に必要な情報を収集して障害解析を行い、被疑セルを論理的にシステムから切り離してシステムを再構成し、各セルの一時停止状態を解除し、システムの運用を続行させる。



【特許請求の範囲】

【請求項 1】 各々メモリと 1 つ以上のプロセッサを含む複数のセルが相互結合網によって相互に接続され、且つ、各セルとサービスプロセッサとが接続されたマルチプロセッサシステムの障害処理方法において、前記メモリを含む共有資源のアクセス時に障害を検出したセルからサービスプロセッサへ障害報告を送出するステップと、

障害報告を受信したサービスプロセッサが全セルの動作を瞬時に停止させるステップとを含むことを特徴とするマルチプロセッサシステムの障害処理方法。

【請求項 2】 サービスプロセッサは、障害報告の受信時に直ちに全セルに対して動作の一時停止を指令することを特徴とする請求項 1 記載のマルチプロセッサシステムの障害処理方法。

【請求項 3】 サービスプロセッサは、受信した障害報告を、セル動作を一時停止する指令としてハードウェア的に折り返して全セルにブロードキャストすることとを特徴とする請求項 1 または 2 記載のマルチプロセッサシステムの障害処理方法。

【請求項 4】 全セルの動作を一時停止させた後、サービスプロセッサが各セルから障害解析に必要な情報を収集して障害解析を行うステップと、障害解析後、サービスプロセッサが被疑セルを論理的にシステムから切り離してシステムを再構成するステップと、システム再構成後、サービスプロセッサが各セルの一時停止状態を解除してシステムの運用を続行させるステップとを、更に含むことを特徴とする請求項 1、2 または 3 記載のマルチプロセッサシステムの障害処理方法。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】 本発明はマルチプロセッサシステムの障害処理方法に関し、より具体的には、各々メモリ及び 1 台以上のプロセッサを含む複数のセルを相互結合網によって相互に接続したマルチプロセッサシステムの障害処理方法に関する。

【0 0 0 2】

【従来の技術】 並列計算機の種類に、メモリ及び少なくとも 1 台のプロセッサを含むセルを複数備え、これら複数のセルをクロスバ網などの高速な相互結合網によって相互に接続したクラスタ計算機と呼ばれるマルチプロセッサシステムがある。この種のマルチプロセッサシステムではセル単位でシステム構成を変更でき、またセル単位で個別のマルチプロセッサシステム（ノード）として独立して運用することができる。

【0 0 0 3】 クラスタ計算機の利点の 1 つは価格性能比に優れていることである。例えば個々のセルとして廉価だが処理能力の高いワークステーションを使用した場合、大型計算機をはるかに凌ぐ価格性能比が得られる。

また他の利点は、共有メモリを物理的に 1 箇所に集中配置する集中共有メモリ型の並列計算機に比べ、システム規模の拡大が容易なことである。更に他の利点は、各々のノードはそれ専用のオペレーティングシステムの制御の下に 1 つの計算機として独立している為、異なるジョブを別々のノードで実行させたり、1 つのジョブを並列プログラムとして同時に複数のノードで実行させたりするなど、多様なジョブ処理形態を取り得ることである。このため、クラスタ計算機は、多種多様のクライアントからの要求に対して迅速な対応が望まれ、また高性能、高信頼性、高可用性が求められるサーバなどに利用される。なお、このようなクラスタ計算機に関する文献としては、例えば特開平 8 - 3 0 5 6 7 7 号公報がある。

【0 0 0 4】 クラスタ計算機は、共有メモリをセル単位ごとにローカルメモリとして分散配置した分散共有メモリ型並列計算機ではあるが、共有メモリ型計算機の種類であるため、プロセッサ間通信モデルは共有メモリモデルに従う。つまり、各セルのプロセッサは、アクセス先のメモリ領域が物理的にどのセルに割り当てられているかの認識はなく、通常のメモリアクセス操作によりアドレス指定で直接的に共有メモリをアクセスすることで、セル間の通信を実現する。具体的には、自セルで発生したメモリアクセス要求が自セルに実装されたメモリに対するアクセスであるときは、自セルのメモリにメモリアクセス要求が転送され、アクセス結果がメモリアクセス元に通知される。他方、自セルで発生したメモリアクセス要求が他セルに実装されたメモリに対するアクセスであるときは、相互結合網を通じて他セルにメモリアクセス要求が転送されて他セルにおいてメモリがアクセスされ、そのアクセス結果が相互結合網を通じて要求元のセルに戻され、そして、メモリアクセス元に通知される。

【0 0 0 5】

【発明が解決しようとする課題】 このようにクラスタ計算機は、セル単位で個別のマルチプロセッサシステム（ノード）として独立して運用できる等の優れた長所を有するが、各セルが相互結合網によって相互に接続され、共有メモリモデルに従ったプロセッサ間通信を行うため、或るセルで障害が発生した場合、その障害が他のセルに伝搬する危険性が高い。例えば、或るセルに実装されたメモリの或る領域に訂正不可能なメモリ障害が発生した場合、一般に計算機を構成する主記憶であるメモリに訂正不可能な誤りが発生したというハードウェア障害は極めて重度の高い障害であるため、その領域をアクセスしたセルは、たとえそのメモリ領域が他セルに実装されているメモリであってもシステム停止障害要因となる。このとき、当該メモリ領域を他のセルが次々とアクセスすると、残りのセルにおいてもシステム停止障害要因が発生する。このため、最悪時には全セルにシステム停止障害要因が発生し、システム全体がダウンする事態を招くこともある。他にも、不正なアドレスを他セルが

3

アクセスすることによってセル間でエラーが伝搬したり、或る 1 つの障害やエラーが原因となって更に別のエラーや障害を引き起こす場合もある。

【0006】このため、クラスタ計算機の運用中において或るセルでエラーや障害が発生した場合、そのセル間の伝搬を速やかに防止する必要があるが、それに有効な技術は未だ提案されていないのが現状である。それ故、障害の伝搬により被疑セルを特定できなかつたり、被疑セルの切り離し、システムの再構成が不可能になる事態を招いていた。

【0007】本発明はこのような事情に鑑みて提案されたものであり、その目的は、システム運用中に或るセルで障害が発生した場合にその障害の伝搬を未然に防止し得るマルチプロセッサシステムの障害処理方法を提供することにある。

【0008】また本発明の別の目的は、システムの信頼性、可用性を向上させることができるマルチプロセッサシステムの障害処理方法を提供することにある。

【0009】

【課題を解決するための手段】本発明は、各々メモリと 1 つ以上のプロセッサを含む複数のセルが相互結合網によって相互に接続され、かつ、各セルとサービスプロセッサとが接続されたマルチプロセッサシステムの障害処理方法において、前記メモリを含む共有資源のアクセス時に障害を検出したセルからサービスプロセッサへ障害報告を送出するステップと、障害報告を受信したサービスプロセッサが全セルの動作を瞬時に停止させるステップとを含むことを特徴とする。より具体的には、サービスプロセッサは、障害報告の受信時に直ちに全セルに対して動作の一時停止を指令する。好ましくは、受信した障害報告を、セル動作を一時停止する指令としてハードウェア的に折り返して全セルにブロードキャストする。

【0010】また、本発明は、全セルの動作を一時停止させた後、サービスプロセッサが各セルから障害解析に必要な情報を収集して障害解析を行うステップと、障害解析後、サービスプロセッサが被疑セルを論理的にシステムから切り離してシステムを再構成するステップと、システム再構成後、サービスプロセッサが各セルの一時停止状態を解除してシステムの運用を続行させるステップとを、更に含むことを特徴とする。

【0011】このように本発明にあっては、メモリや入出力装置などの共有資源のアクセス時に、或るセルにおいて訂正不可能なメモリ障害や不正なアドレスに対するアクセス等の障害が検出されると、そのセルからサービスプロセッサへ障害報告が出され、この障害報告を受信したサービスプロセッサが全セルの動作を瞬時に停止させるため、障害がセル間で伝搬するのを未然に防止することができる。そして、全セルの動作を一時停止させた後、障害解析に必要な情報の収集と解析を行って被疑セルを特定し、被疑セルを論理的にシステムから切り離し

4

てシステムを再構成した後、各セルの一時停止状態を解除してシステムの運用を続行させることで、システムの信頼性、可用性を向上させることができる。

【0012】

【発明の実施の形態】次に本発明の実施の形態の例について図面を参照して詳細に説明する。

【0013】図 1 を参照すると、本発明を適用したマルチプロセッサシステムの一例は、複数のセル 400 をネットワーク（相互結合網）500 により相互接続することにより構成される。ここで、システムは 4 つのセル 400 を有するものとし、それぞれのセル 400 はそれぞれ 4 つのプロセッサ 200 およびメモリ（メモリコントローラを含む）300 と、セル制御回路 100 を有するものとする。但し、この数はシステムの要求に応じて任意に設定することができる。また、各セル 400 は専用線 700 によってサービスプロセッサ 600 に接続されている。

【0014】メモリ 300 は、各セル 400 に分散して存在しており、プロセッサ 200 から見ると、自セル内のメモリと他セルのメモリとが存在することになる。しかし、プロセッサ 200 はアクセス先のメモリ領域が物理的にどのセルに割り当てられているかの認識はなく、通常のメモリアクセス操作によりアドレス指定で直接的にメモリ 200 をアクセスする。即ち、プロセッサ間通信モデルは共有メモリモデルに従う。この共有メモリモデルをサポートしているのが、本実施例ではセル制御回路 100 である。

【0015】図 2 を参照すると、各セル内のセル制御回路 100 は、アドレスレジスタ 110 と、アドレス解決テーブル 120 と、ライト保護フラグ 130 と、セル番号レジスタ 141 と、アクセス種別レジスタ 142 と、比較器 150 と、論理積回路 160 とを含んでいる。

【0016】アドレス解決テーブル 120 は、システムの立ち上げ時にサービスプロセッサ 600 により初期設定される。このアドレス解決テーブル 120 により、各セルに分散して存在するメモリ 300 はアドレスが重複しない単一のメモリ空間に構成される。プロセッサ 200 が要求したメモリアドレスは、アドレス解決テーブル 120 を索引することにより、どの物理セルへのアクセスかが判明する。アドレス解決テーブル 120 は、複数のエントリを有し、アドレスレジスタ 110 に保持されたプロセッサ 200 またはネットワーク 500 からのアドレスの中のモジュールアドレス 111 により索引される。各エントリは、有効ビット 121、セル番号 122 及びセル内モジュール番号 123 を保持する。有効ビット 121 は、当該エントリが有効か否かを指示する。例えば“0”であれば有効でなく、“1”であれば有効であることを指示する。セル番号 122 は、当該アドレスに相当するメモリモジュールが存在するセルの番号を表示する。セル内モジュール番号 123 は、当該アドレス

に相当するメモリモジュールがセル内の何れのメモリ 300 であるかを表示する。セル内モジュール番号 123 とモジュール内アドレスオフセット 112 とを繋げたものがセル内アドレス 191 となる。

【0017】ライト保護フラグ 130 は、他のセルからのアクセスがライトであった場合にそのライトを許可するか否かを指示する。例えば、“0”のときは他のセルからのライトを許容するが、“1”のときは他のセルからのライトは許容せずにアクセス例外とする。

【0018】セル番号レジスタ 141 は、アクセスリクエストを発行したプロセッサ 200 の存在するセルのセル番号を格納するためのレジスタである。アクセス種別レジスタ 142 はアクセスリクエストの種別を表す。例えば、“1”の場合にライトを意味する。比較器 150 は、セル番号レジスタ 141 の内容とアドレス解決テーブル 120 から読み出したセル番号 122 とを比較する。論理積回路 160 は、アドレス解決テーブル 120 の有効ビット 121 が有効を示し、アクセス種別がライトであり、ライト保護フラグ 130 がライトを許容せず、且つ、アドレス解決テーブル 120 から読み出したセル番号 122 がセル番号レジスタ 141 の値と一致しない場合に、アドレス例外発生信号 161 を出力する。

【0019】次に、プロセッサ 200 によるメモリアクセス時の動作について説明する。

【0020】図 1 から図 3 を参照すると、プロセッサ 200 からメモリアクセスがあると、モジュールアドレス 111 によってアドレス解決テーブル 120 を索引する（図 3 のステップ S1）。その結果、有効フラグ 121 が無効を示していれば（ステップ S2）、当該アクセスは存在しないアドレスにされたものであるとしてアドレス不在例外とする。また、他のセルに存在するメモリアドレスであることが判明すると（ステップ S3）、ネットワーク 500 を介して当該他のセルでアクセスが行われる。また、自セルに存在するメモリアドレスであれば、自セル内の該当するメモリモジュールにアクセスする（ステップ S4）。

【0021】また、他のセルからアクセスがあった場合、そのアクセスがライトでなければ（ステップ S7）、自セルの場合と同様に該当するメモリモジュールにアクセスする（ステップ S4）。これに対し、他のセルからのアクセスがライトの場合にはライト保護フラグ 130 をチェックし（ステップ S8）、他のセルからのライトを許容する旨を指示していれば該当するメモリモジュールにアクセスし（ステップ S4）、他のセルからのライトを許容しない旨を指示していればアクセス例外とする。

【0022】該当メモリモジュールにアクセスした結果、例えばパリティエラーや訂正不可能なメモリ障害が発生した場合（ステップ S5）、専用線 700 を介してサービスプロセッサ 600 に障害報告を送出する（ステ

ップ S6）。アドレス不在例外、アクセス例外が発生した場合も、専用線 700 を介してサービスプロセッサ 600 に障害報告を送出する（ステップ S6）。

【0023】次に、障害報告を受けたサービスプロセッサ 600 の動作を説明する。

【0024】図 4 を参照すると、障害報告を受けたサービスプロセッサ 600 は全てのセル 400 の処理を直ちに一時停止させる（ステップ S11）。その実施例を図 5 に示す。図 5 を参照すると、各セル 400 から出された障害報告を伝達する信号線 701 はサービスプロセッサ 600 内のプロセッサ 601 の割り込み端子に接続されると共に、折り返し回路 602 を構成する論理和回路 603 の入力端子にも接続されている。また論理和回路 603 の出力は、各セル 400 の各部に対して動作を一時停止すべき指令を伝達する信号線 702 に接続されている。これによって、信号線 701 に障害報告が出されると、瞬時に全セルに対してセル動作を一時停止する指令が信号線 702 によってブロードキャストされる。また、プロセッサ 601 は信号線 702 によって起こる割り込みによって障害報告を認識でき、また、信号線 702 の状態を制御することにより、一時停止の指令を解除することができる。なお、専用線 700 を構成する他の信号線群 703 は、サービスプロセッサ 600 がシステムの初期設定、構成変更時の設定、障害情報の採取などに使用する信号線群である。

【0025】サービスプロセッサ 600 は、次に、停止状態の全てのセル 400 から、障害解析に必要な各種の情報（例外発生アドレス、障害発生アドレス、例外や障害の種類、一時停止直前に発行されたトランザクションの情報、プロセッサの内部レジスタ情報など）を障害情報として採取し（ステップ S12）、どのような障害がどのセルで発生したか、その障害の影響範囲などを調べる障害解析を行う（ステップ S13）。そして、障害の起因であるセル（被疑セル）を特定すると、被疑セル及びそのセルを含むノードを運用から切り離してシステムが運用可能かを判断し（ステップ S14）、可能であれば被疑セル及びそのセルを含むノードをシステムから論理的に切り離し（ステップ S15）、残りのセルでシステムを再構成する（ステップ S16）。そして、構成変更の設定が終了しシステムが被疑セル及び被疑ノード無しで運用できる状態になると、セル 400 に対する一時停止の指令を解除する（ステップ S17）。これにより、システムは被疑セル及び被疑ノードが切り離された状態で運用を再開する。他方、被疑セル及び被疑セルを含むノード以外に正常なノードが残っていない場合など、被疑セル及びそのセルを含むノードを運用から切り離すとシステムの運用が不可能になる場合は、ステップ S15～S17 の処理はスキップする。この場合、システムは停止したままとなる。

【0026】次に本実施の形態の具体例について説明する。

【0027】4つのセル400（\$0～\$3とする）のアドレス解決テーブル120の設定を図6のようにした場合、図7のメモリ構成が実現される。図7において、実線部分は各セルに物理的に搭載されたメモリである。ここでは全セルが同量のメモリを備えると仮定しているが、実際は同じでなくて良い。また、縦軸はそれぞれのセルから見たメモリアドレスを示し、全ノードとも0番地から始まる空間を持つと仮定している。

【0028】このメモリ構成では、セル\$0とセル\$1が対象型マルチプロセッサ構成の独立したコンピュータを構成し、2つで1つのノード#0として、セル\$2（ノード#2）、セル\$3（ノード#3）とともに図8に示されるような3ノードのクラスタ計算機を構成する。ノード#0においては、セル\$0の物理メモリ全部とセル\$1のメモリモジュールx0からx2の合計7モジュールが、ローカルメモリとしてセル\$0及びセル\$1から共通にアクセスできるように構成されている。セル\$1のメモリモジュールx3は、クラスタ共有メモリ（通信エリア）としてノード間で共有されている。また、セル\$2及びセル\$3のメモリモジュールx0からx2は、それぞれのノードのローカルメモリにマップされ、互いに独立である（セル固有メモリ）。一方、セル\$2のメモリモジュールx8とセル\$3のメモリモジュールx9の2モジュールが共有通信エリアに設定されている。

【0029】この図6の設定の場合、ライト保護フラグ130は、特定のセル（同じノードを構成するセル）からのライトアクセスを許容するよう設定される。

【0030】次に、図8乃至図11を参照して、本実施例における障害処理の具体例を説明する。

【0031】図8において、ノード#0のセル\$1の或るプロセッサ200が、同じノード#0のセル\$0のメモリモジュールx3に対してメモリアクセス要求を出したとすると、その要求は矢印aに示すルートで処理されて、セル\$0においてメモリモジュールx3がアクセスされる。このとき、メモリモジュールx3で例えば訂正不可能なメモリ障害が発生したとすると、それを検出したセル\$0からサービスプロセッサ600に障害報告が出される（図8のb）。この障害報告はハードウェア的に折り返され、一時停止指令として全セル\$0～\$3に伝達される（図8のc）。これによって、全セル\$0～\$3はその動作を一時停止する。その後、サービスプロセッサ600は、全セル\$0～\$3から障害情報を採取し（図9のd）、障害解析を行う。そして、セル\$1からのメモリアクセス要求時に同じノード#0のセル\$0のメモリモジュールx3に訂正不可能なメモリ障害が発生したこと、その影響はノード#0内に限られることを判別し、被疑セル\$0を含むノード#0をシステムから

切り離し、ノード#2、#3だけで運用を続ける。

【0032】他方、図10において、ノード#2のセル\$2の或るプロセッサ200が、ライト保護がなされているノード#0のセル\$1におけるメモリモジュールx7に対してライトアクセスを行った場合（図10のa）、セル\$1においてアクセス例外が発生し、セル\$1からサービスプロセッサ600に障害報告が出される（図10のb）。この障害報告はハードウェア的に折り返され、一時停止指令として全セル\$0～\$3に伝達される（図10のc）。これによって、全セル\$0～\$3はその動作を一時停止する。その後、サービスプロセッサ600は、全セル\$0～\$3から障害情報を採取し（図11のd）、障害解析を行う。そして、ノード#2のセル\$2から、ライト保護されているノード#0のセル\$1のメモリモジュールx7にライトアクセスが行われたことによりアクセス例外が発生したこと、その影響はノード#2内に限られることを判別し、被疑セル\$2を含むノード#2をシステムから切り離し、ノード#0、#3だけで運用を続ける。

【0033】以上の実施の形態では、各セル内のプロセッサ200とメモリ300とがセル制御回路100を介して接続される構成を採用したが、図12に示されるように、プロセッサ200とメモリ300とがプロセッサバス801で接続される構成であっても良い。この場合、セル制御回路100と同機能を有するディレクトリ802をプロセッサバス801に接続する。また、以上の実施の形態では、各セルに入出力コントローラを含めなかったが、図12に示されるように入出力コントローラ803がプロセッサバス801に接続されていても良い。この場合、入出力コントローラ803に対するアクセスもメモリ300と同じように行われる。つまり、各プロセッサ200は入出力コントローラが物理的にどのセルに割り当てられているかの認識なく、通常の入出力操作によりアドレス指定で直接的に入出力コントローラをアクセスすることで、セル間の通信を実現する。具体的には、ディレクトリ802は、自セルで発生したI/O要求が自セルに実装された入出力コントローラに対するものであるときは、自セルの入出力コントローラにI/O要求を転送し、結果を要求元に通知する。他方、自セルで発生したI/O要求が他セルに実装された入出力コントローラに対するものであるときは、相互結合網を通じて他セルにI/O要求を転送し、他セルにおいて入出力コントローラがアクセスされ、その結果が相互結合網を通じて要求元のセルに戻され、そして、要求元に通知される。

【0034】

【発明の効果】以上説明したように本発明によれば、システム運用中に或るセルで障害が発生した場合にその障害の伝搬を未然に防止することができる。その理由は、障害が発生した時、その報告を受けたサービスプロセッ

サが全セルの動作を瞬時に停止させるからである。

【0035】また本発明によれば、システムの信頼性、可用性を向上させることができる。その理由は、サービスプロセッサが、全セルの動作を一時停止させた後、障害解析に必要な情報の収集と解析を行って被疑セルを特定し、被疑セルを論理的にシステムから切り離してシステムを再構成した後、各セルの一時停止状態を解除してシステムの運用を続行させるからである。

【図面の簡単な説明】

【図1】本発明を適用したマルチプロセッサシステムの一例を示すブロック図である。

【図2】セル制御回路の実施例のブロック図である。

【図3】本発明の実施の形態における動作を説明するための流れ図である。

【図4】サービスプロセッサの動作を説明するための流れ図である。

【図5】障害報告を受けたサービスプロセッサが全セルを直ちに一時停止させる実施例のブロック図である。

【図6】アドレス変換テーブルの設定例を示す図である。

【図7】メモリ構成の一例を示すメモリマップである。

【図8】本発明を適用したマルチプロセッサシステムにおける障害処理の具体例の説明図である。

【図9】本発明を適用したマルチプロセッサシステムにおける障害処理の具体例の説明図である。

【図10】本発明を適用したマルチプロセッサシステムにおける障害処理の具体例の説明図である。

【図11】本発明を適用したマルチプロセッサシステムにおける障害処理の具体例の説明図である。

【図12】本発明を適用したマルチプロセッサシステムの別の例を示すブロック図である。

【符号の説明】

100…セル制御回路

200…プロセッサ

300…メモリ

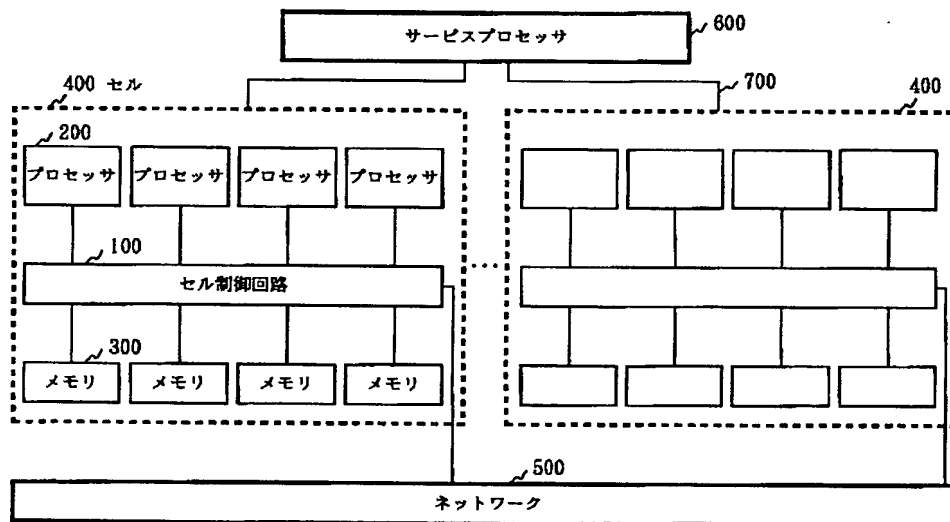
400…セル

500…ネットワーク

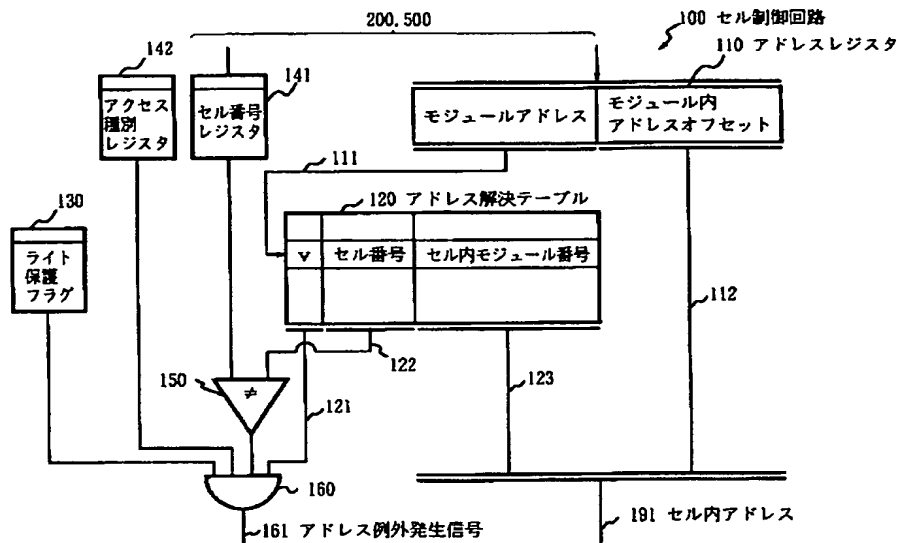
600…サービスプロセッサ

700…専用線

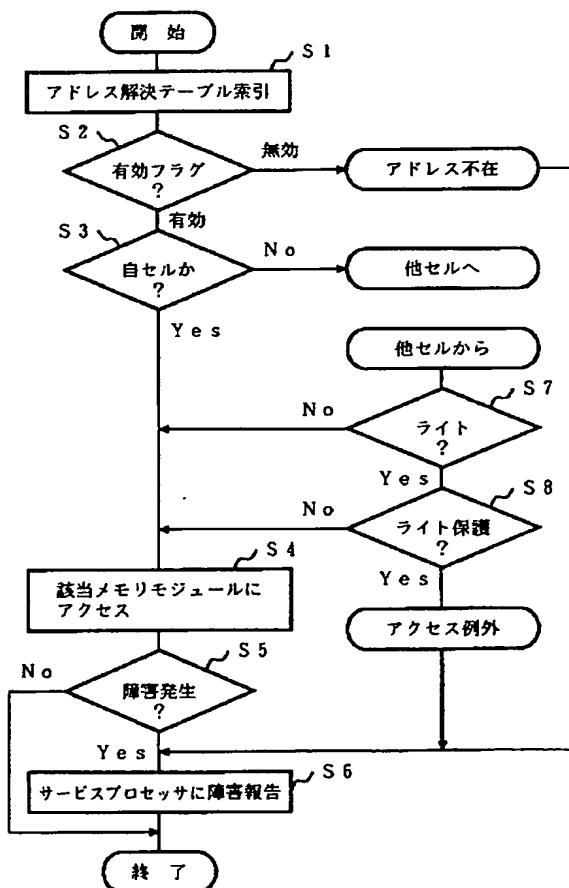
【図1】



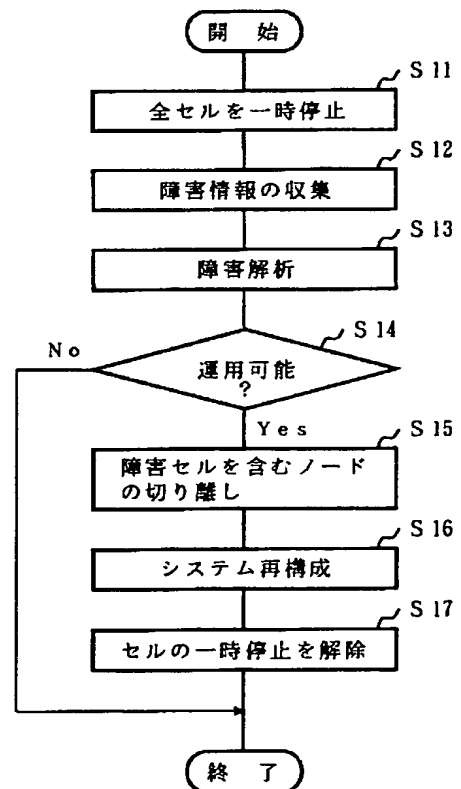
【図2】



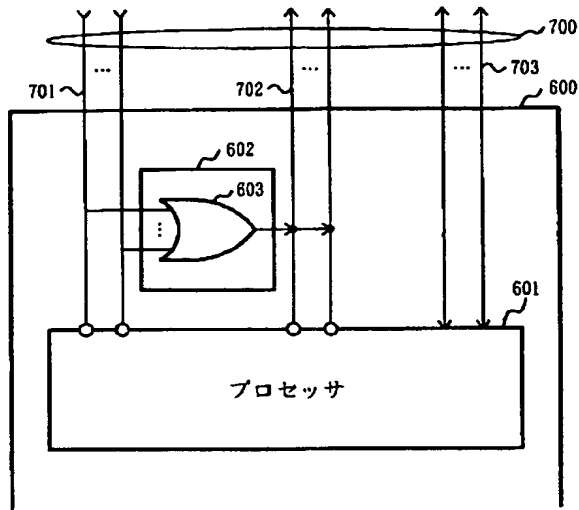
【図3】



【図4】



【図 5】



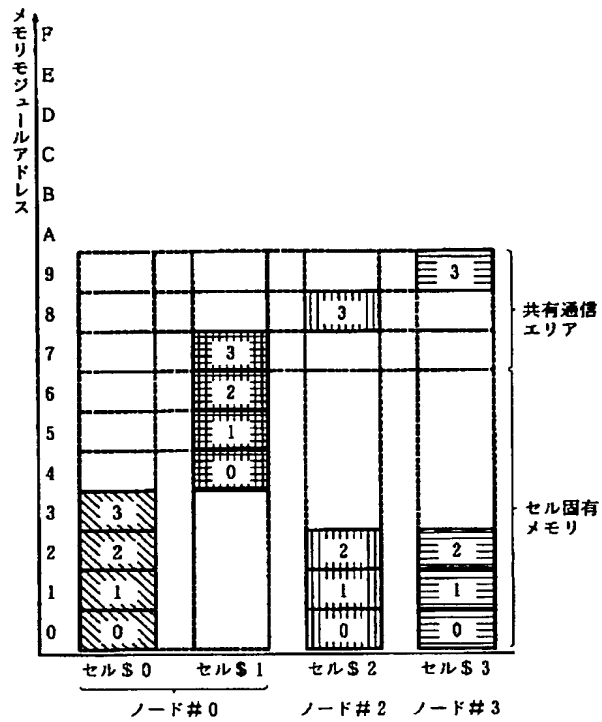
【図 6】

0	1	0	0	0	1	0	0	0	1	2	0	0	1	3	0
1	1	0	1	1	1	0	1	1	1	2	1	1	1	3	1
2	1	0	2	1	0	2	1	2	2	2	1	3	2	3	2
3	1	0	3	1	0	3	0	—	—	—	0	—	—	—	—
4	1	1	0	1	1	0	0	—	—	—	0	—	—	—	—
5	1	1	1	1	1	1	0	—	—	—	0	—	—	—	—
6	1	1	2	1	1	2	0	—	—	—	0	—	—	—	—
7	1	1	3	1	1	3	1	1	3	3	1	1	3	3	3
8	1	2	3	1	2	3	1	2	3	3	1	2	3	3	3
9	1	3	3	1	3	3	1	3	3	3	1	3	3	3	3
A	0	—	—	0	—	—	0	—	—	—	0	—	—	—	—
B	0	—	—	0	—	—	0	—	—	—	0	—	—	—	—
C	0	—	—	0	—	—	0	—	—	—	0	—	—	—	—
D	0	—	—	0	—	—	0	—	—	—	0	—	—	—	—
E	0	—	—	0	—	—	0	—	—	—	0	—	—	—	—
F	0	—	—	0	—	—	0	—	—	—	0	—	—	—	—

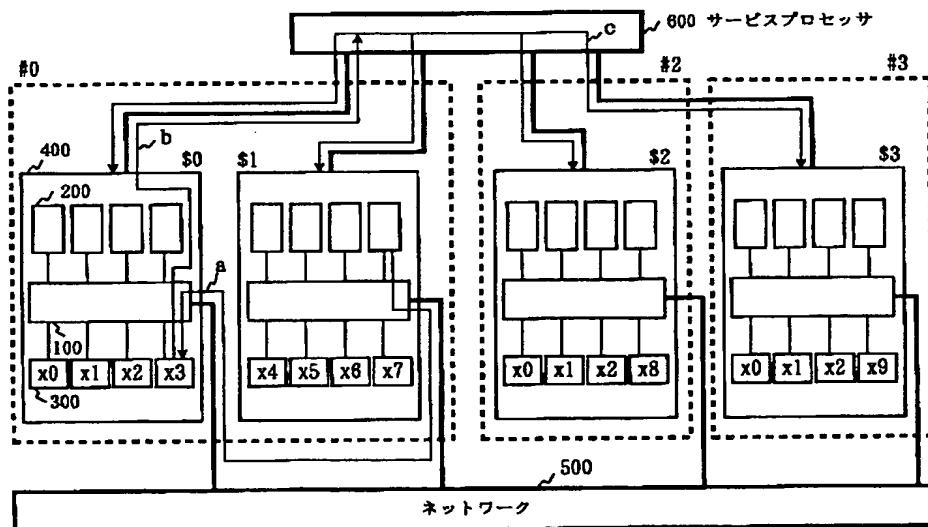
セル\$ 0 セル\$ 1 セル\$ 2 セル\$ 3

ノード#0 ノード#2 ノード#3

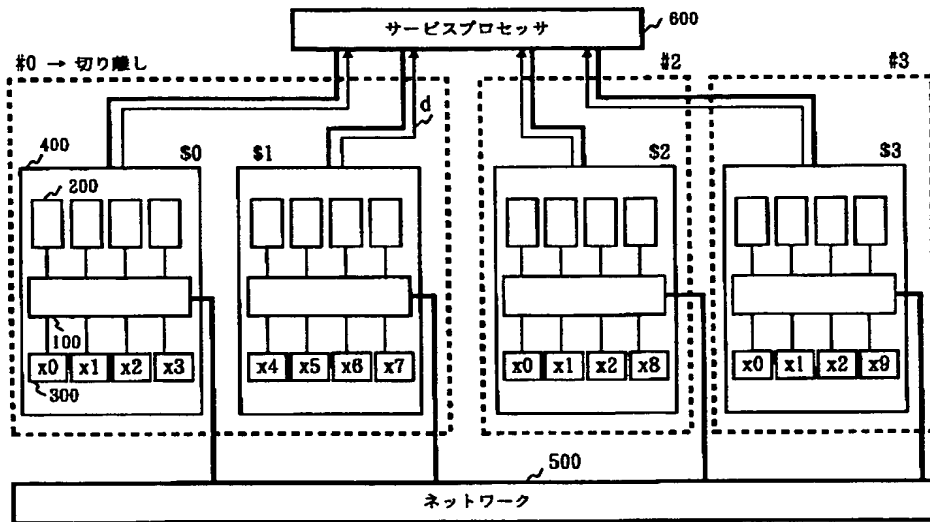
【図7】



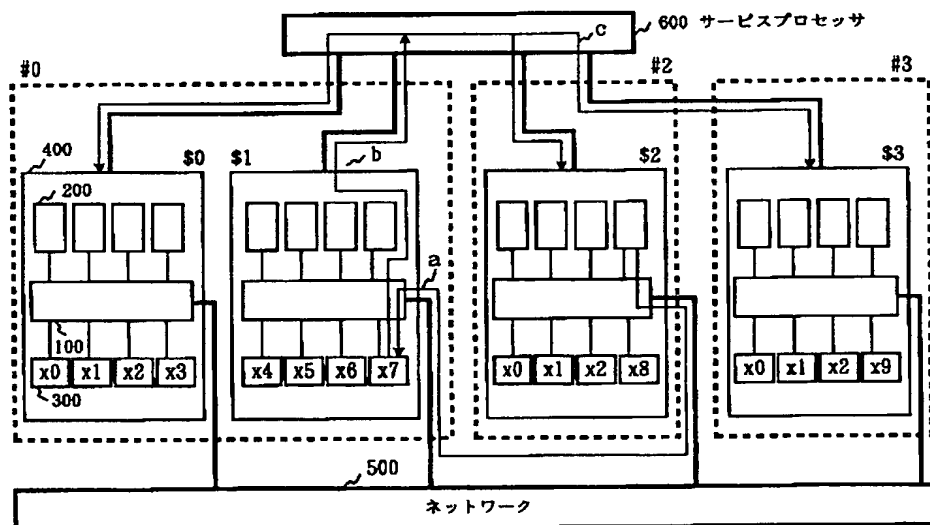
【図8】



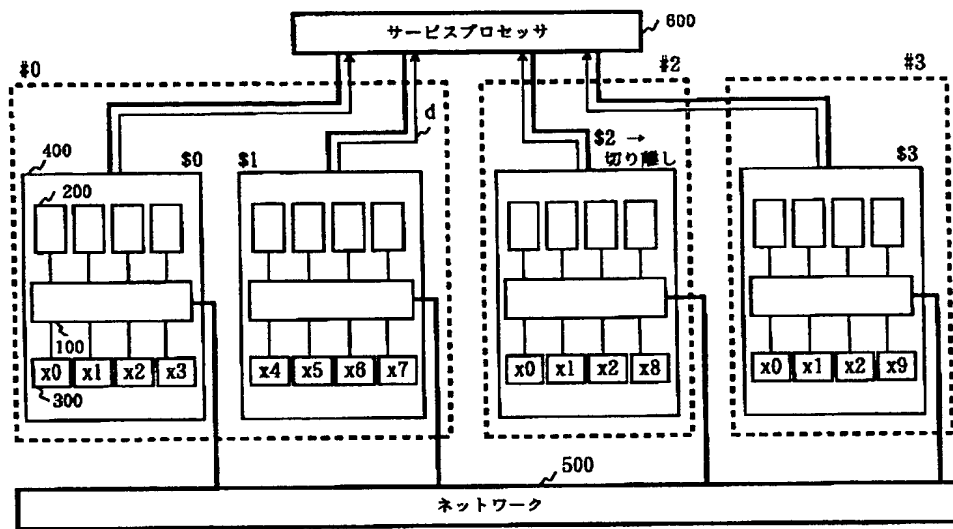
【図 9】



【図 10】



【図11】



【図12】

